8

UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/020,697	10/29/2001	Mary Lynn Cashin	RPS920010135US1	1333

45211 7590 03/09/2005

KELLY K. KORDZIK
WINSTEAD SECHREST & MINICK PC
PO BOX 50784
DALLAS, TX 75201

EXAMINER	
VO, TED T	
ART UNIT	PAPER NUMBER
2122	

DATE MAILED: 03/09/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No. 10/020,697	Applicant(s) CASHIN ET AL.	
	Examiner Ted T. Vo	Art Unit 2122	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on 25 October 2004.

2a)☒ This action is **FINAL**. 2b)□ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) 1-15 is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) 1-4,6-9 and 11-14 is/are rejected.

7)☒ Claim(s) 5,10 and 15 is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)□ The drawing(s) filed on _____ is/are: a)□ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) □ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date 10/29/01.

4) □ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application (PTO-152)

6) □ Other: _____.

DETAILED ACTION

1. This action is in response to the amendment filed on 10/25/2004 responsive to the Office action dated, 08/26/04.

- Claims 1-15 remains original filed.

Applicants' arguments have been fully considered. Within this detailed action,

- Claims 1-4, 6-9, 11-14 stands finally rejected under 35 U.S.C. 102(e) as being anticipated by Teoman et al. (US No. 6,370,614 B1).

- Claims 5, 10, and 15 are objected to as being direct to allowable subject matter.

- Claims 1-15 are pending in this application.

Response to Arguments

2. Applicants' arguments in the Remarks section filed on 10/20/2004 have been fully considered.

Applicants argue that Teoman does not disclose, "*defining a particularly preload object with one or more distribute*" as recited Claims 1 and similarly in Claims 6 and 11: (Remarks, page 2, lines 15-17),

by alleging that:

> Figure 6 of Teoman discloses file and driver objects Teoman instead discloses file and driver objects that are used, in at least one embodiment, to determine the sequence in which device drivers are called to service an 1/0 request. Figure 10 of Teoman instead discloses an exemplary user interface generated by the user cache manager to permit the user to configure memory management and preloading policies for the user cache. Teoman further discloses that examples of preloading policies include preloading complete files in response to tile segment access, preloading all tiles within the directory or folder of a launched application, preloading all files in a directory or folder if a threshold number of tiles from the directory or folder have already been accessed, preloading tiles in the system directory or folder, preloading files having a particular tile type identifier if a threshold number of files having the file type identifier have been accessed, and so forth.
> (Remarks, page 2, lines 19-21)

Examiner respectfully disagrees: In FIG. 6, OBJECT FILE and DRIVER OBJECTs have functionality/means corresponding to claimed limitation "defining preload object". For viewing the operation of FIG. 6, it would require considering the whole reference as well.

At first, as shown by FIG. 1, Teoman teaches "preloading" (as described in FIG. 6 and FIG. 10) that is used in order to write files into a user cache (25). This teaching is in the same manner as in the Applicants' specification (In the Applicants' specification, it uses a preload object to install "device driver" (206) to load on to a storage unit: "*a preload which may comprise one or more of a device driver, operating system and application software may be loaded onto a storage unit, e.g., disk unit 20, of a computer system, e.g., data processing system 100, before the computer system is actually manufactured*") (See Applicants' specification: Figure 2 and its associated text, page 9, lines 16-23)).

- Referring FIG. 10 of the reference, it shows a User Cache in which a user can enter user preferences for "preloading" the files. Its purpose is **preloading** the files/data from mass storage to a user cache. FIG. 10 includes allowing a limit the number of files loaded into the system memory. **Preloading** the files, could be seen via FIGs. 3 and 5 as well, and the associated texts; the teaching shows how the files installed/written into system memory/user cache from a mass storage 46 by using preloading (See Column 10, lines 16-27).
- Referring FIG. 6 of the reference, it discloses, "defining preload object" in association with preloading of FIG. 10. Indeed, in FIG. 6, Teoman discloses an **instantiation of FILE OBJECT 105** that includes hierarchy of DRIVER OBJECTs 107, 109, 111 (See column 10, lines 63-66: "*When a file 15 is first opened for access, the I/O manager (e.g., element 43 of FIG. 5) **<u>instantiates</u>** a file object 105 in system memory to represent the file 15*"). **FILE OBJECT 105** is **instantiated** and clearly represents to a FILE 15 in massage storage. Thus, "instantiating" of FILE OBJECT in "preloading" operation via FIG. 6 and FIG. 10 discloses that FILE OBJECT 105 (including DRIVER OBJECTs 107, 109, 111) is functionally equivalent to this recited limitation, "*defining a particularly preload object with one or more distribute*".

Applicants argue that Teoman does not disclose *"comparing attributes of said one or more software element objects with said one or more attributes of said particular preload object, wherein each of said one or more software element objects constitutes one or more of a device driver object, an operating system object and an application software object" as* recited in Claim 1 and similarly in claims 6 and 11 (Remarks, page 3, lines 9-13), by alleging that

> Teoman instead discloses data integrity checking which may encompass having the data in the user cache being periodically compared against the corresponding data in the mass storage device. Column 14, lines 61 -63. Teoman further discloses that if the data does not match, the operating system or user (or both) may be altered so that an archive of the correct data may be generating. Column 14, lines 63-66. Comparing data in the user cache against the corresponding data in the mass storage data is not the same as comparing attributes of a software element object with attributes of a preload object.
> (Remarks, page 3, lines 16-23)

Examiner respectfully disagrees: It should be noted that "*data integrity checking*" is part of data caching, and included with preload operation. For example, Teoman discloses, "*Because data stored in user cache corresponds to and should be a duplicate of data stored in a mass storage device, data check integrity can be implement*" (Column 14, lines 59-61). Data caching requires accessing data (FILE 15) in Mass Storage. As shown in FIG. 6, the **instantiation** of FILE OBJECT 105 in user computer (FILE OBJECT 105 represents FILE 15 in mass storage) includes a hierarchy of DRIVER OBJECTs and attributes. And checking data integrity is done periodically in association with preloading. It should be noted that caching and checking can not separated, and it should be noted that the FILE OBJECT 105 is **instantiated** at user system representing caching data at the remote or local mass storage. As addressed, FILE OBJECT 105 comprises attributes and DRIVER Objects, thus data (FILE 15) comprises attributes and drivers (because FILE OBJECT represents FILE 15). The accessing data is performed via preloading FILE OBJECT 105 by I/O request service. To detail how the data (FILE 15) can be compared and accessed by the I/O request service in FIG 6, Teoman illustrates FIG. 7. Particularly, comparing is discussed in Column 12, lines 1-57. For example, to disclose comparing, Teoman picks a portion DRIVER OBJECT 109 and USER CACHE DRIVER 45 in FIG. 6, where Teoman describes, "*access packet 114 that include a handle to the file object and values indicating the amount of data to be accessed*" (Column 12, lines 2-4), and "*The user cache driver 45 **compares the blocks** indicated by the access packet 114 to the directory of the user cache 25*" (column 12, lines 18-21). It should be noted that within the illustration in FIG. 6, it visually shows such blocks, as "ATTRIBUTE's 108, 110, 112. Thus, when data (FILE 15) used in the user cache is periodically compared against the corresponding data (FILE 15) in a remote/local mass storage (Column 14, lines 61-63), it acts comparing as within in the mechanism shown in FIG. 6. It is reemphasized once that showing in FIG 6, the FILE OBJECT ("*preload* *object*") represents to data/FILE 15 and comprises ATTRIBUTEs and hierarchy of DEVICE OBJECTs for teaching claiming, "*comparing attributes of said one or more software element objects* (within FILE 15) *with said one or more attributes of said particular preload object* (within FILE OBJECT 105), *wherein each of said one or more software element objects constitutes one or more of a device driver object* (within DEVICE OBJECTs), *an operating system object and an application software object*", as recited in Claim 1 and similarly in Claims 6 and 11".

In the Remark, within pages 3 and 4 (page 3, last four lines, pages 4, lines 1-8), particularly, Applicants address "ex parte Levy" and "re Robertson" and require evidences for comparing data in *user cache and mass storage device* discloses comparing attributes of software element object with attribute of preload object.

Examiner would respond: "ex parte Levy" relates to the lacking inherency based on objective evidence or cogent technical reasoning, and in "re Robertson" it is ruled based on the missing "third fastening element" that is *insufficient* for inherency. However, Applicant should also be noted that in In re Schreiber, 128 F.3d 1473, 44 USPQ2d 1429 (Fed. Cir. 1997), the court affirmed a finding that a *conical spout* used primarily to dispense oil from an oil can inherently performed to a conical container top for dispensing popped popcorn based on structure similarity (emphasis added).

By contrast to the Applicants' addressing above, this case Teoman discloses all elements and performs preloading as in light of the Applicants' specification. Particularly, Teoman shows FIG. 6 and addresses data integrity in column 14, lines 61-64. The evidence is clearly that accessing "data" (FILE 15) for caching/storing and comparing "data" (FILE 15) is done via the instantiation of FILE OBJECT 105 including integrity checking as mentioned in Column 14, lines 61-64.

Applicants argue that Teoman does not disclose, *"identifying one or more of said one or more software element objects whose attributes comprise said one or more attributes of said particular preload object"* as recited in claim 1 and similarly in claims 6 and 11 (Remarks, page 4, lines 16-18) by alleging that

> "Figure 6 of Teoman instead discloses file and driver objects that are used, in at least one embodiment, to determine the sequence in which device drivers are called to service an 1/0 request. There is no language in the Teoman that discloses that these file and driver objects comprise attributes of a preload object".
> (Remarks, page 4, lines 21-24)

Examiner respectfully disagrees: It should be noted that FILE OBJECT 105 is functionally equivalent to "Preload object", and FIG. 6 shows that FILE 15 in mass storage is identified/represented by FILE OBJECT 105 (Preload object) in user cache as addressed above. It should be further considering FIG. 11. FIG. 11 shows choosing specific files to preload, where in FIG. 11, the identified file/data (with file type such as .exe, .doc, etc) is corresponding to its path and size. It should be noted that "attribute" is a generic term defined as a property of an object or may be considered a container for the property of the object. Therefore, data types such as '.exe', '.doc', and the file size, etc., are also having attribute means as well.

Applicants argue that Teoman does not disclose *"installing software associated with said identified one or more software elements objects onto a particular preload associated with said particular preload object"* as recited in claim 1 and in similarly in claim 6 and 11 (Remarks, page 4, lines 27-29) by alleging that

> Teoman instead discloses examples of preloading policies (Remarks page 5, lines 2-4),
> Teoman discloses the user may select files a indicate a particular file is to be <u>preload into a cache</u> (Remarks page 5, lines 10-11),
> Figure 12 of Teoman instead discloses an exemplary user interface generated by the user cache manager to permit the user to generate a report of the user cache contents, test the memory in the user cache, flush the user cache or transfer the contents of the user cache to a mass storage such as a tape backup (Remarks page 5, lines 12-15).

Examiner disagrees: As addressed, OBJECT FILE 105 is carried out in an instantiation for data caching. In the Teoman's summary (column 2, lines 1-10), its disclosure includes caching data from mass storage to user cache ("installing"). This does the same as in the Specification (Specification: Figure 2 and its associated text, page 9, lines 16-23) as addressed above. As addressed, the acts of caching (installing) are done by I/O request service and shown by FIG. 6. Furthermore, see Column 8, lines 65-67, "Once data from remote database is cached in user cache", and FIG. 11 is another exemplary for showing some identifying files to be cached/loaded from various locations into the user cache (Column 16, lines 1-37. For example, "*The user may then double click <u>selected logical drivers, directories or filenames</u> to indicate that files meeting the specifying criteria are to be preloaded*"). All of these have "installing" means.

Applicants argue that Teoman does not disclose *"modifying an attribute of said identified one or more software element objects to match said one or more attributes of said particular preload object"* as recited in claim 2 and similarly in claims 7 and 12 (Remarks, page 5, lines 27-29), by alleging that Figure 10 of Teoman instead discloses an exemplary user interface generated by the user cache manager to permit the user to configure memory management and preloading policies for the user cache.

Examiner disagrees: Modifying means in such a claim could be done by a user who accesses the user cache. In the whole context of this reference, Teoman shows the configuration acted with "USER". For example, column 9, lines 65-66, a user can write "C:\SYSTEM\" in preloading. It means that a user can do anything via the interface. FIG. 6 shows comparing and identifying data using attributes as addressed. FIG 10 shows extension of matching files could be changed. FIG. 11 shows a user selecting data type, or excluding it. And Checking data integrity (Column 14) lines 59-67) could be alerted to a user so that correct data could be generated. All things like these disclose *"modifying an attribute of said identified one or more software element objects to match said one or more attributes of said particular preload object"*.

Regarding the Applicants arguments to Claims 3, 8, 13, (Remarks: page 6, lines 12-26) and Claims 4, 9, 14, where Applicants traverse the Examiner citations, and arguing that:

"there is no language in the cited passage or in the description of Figure 6 of Teoman that discloses a software element object associated with attribute data where the <u>attribute data includes either operating system information or installation information</u> (Remarks: page 6, lines 21-24) and,

"there is no language in Teoman that discloses that the attributes of a driver object includes a part number (Remarks: page 7, lines 5-7).

Examiner responds:

First of all, the Claims include limitation "includes either operating system information or installation information" (addressed to dependent Claims: 3, 8, 13) and "includes a part number" (addressed to dependent Claims: 4, 9, 14) are all non-functional descriptive materials within the scope of these Claims because these limitations do not impart functionality within the scope of the Claims. These limitations would be identified as non-functional descriptive materials.

Applicants are respectfully referred to MPEP 2106:

> *In re Dembiczak,* 175 F.3d 994, 1000, 50 USPQ2d 1614, 1618 (Fed. Cir. *1999). Nonfunctional* descriptive material cannot render nonobvious an invention that would have otherwise been obvious. Cf. *In re Gulack,* 703 F.2d 1381, 1385, 217 **USPQ** 401, 404 (Fed. Cir. 1983) (when descriptive material is not functionally related to the substrate, the descriptive material will not distinguish the invention from the prior art in terms of patentability)" (emphasis added).
>
> MPEP 2106 IV B 1 (b) indicates that "nonfunctional descriptive material" is material "that cannot exhibit any functional interrelationship with the way in which computing processes are performed".

Examples of "nonfunctional descriptive material" include certain types of "printed matter".

> MPEP 2112.01 indicates that "Where the only difference between a prior art product and a claimed product is printed matter that is not functionally related to the product, the content of the printed matter will not distinguish the claimed product from the prior art. In re Ngai, ___ F.3d ___, 2004 WL 1068957 (Fed. Cir. May 13, 2004)(Claim at issue was a kit requiring instructions and a buffer agent. The Federal Circuit held that the claim was anticipated by a prior art reference that taught a kit that included instructions and a buffer agent, **even though the content of the instructions differed**.). See also In re Gulack, 703 F.2d 1381, 1385-86, 217 USPQ 401, 404 (Fed. Cir. 1983)("Where the printed matter is not functionally related to the substrate, the printed matter will not distinguish the invention from the prior art in terms of patentability .... [T]he critical question is whether there exists any new and unobvious functional relationship between the printed matter and the substrate."

In this case, claiming "operating system information", "installation information", or "a part number", where these claimed materials do not impart any functionality, would not make these descriptive materials be different from the "attribute" or "data" shown within FIG. 6 of Teoman. These claiming descriptive materials are the same as in the "printed matter" case.

Particularly, at Columns 6-10, Teoman describes general features shown in FIGs 4-5 and that discloses **broad means** "operating system **information**", "installation **information**". FIG. 6 (pointing to Object Drivers, and File 15 in Mass storage) also has such **broad means** (a part number as a content in FILE 15).

- Regarding Claims 3, 8, and 13. Teoman shows FIG 6, with blocks 108, 110, 112 used to access derivers directly to FILE 15 in user cache/mass storage. Furthermore, Teoman shows OS 30, DECIVES DRIVERs 32, APPS 34, and OC CACHE 36 in a SYSTEM MEMORY in FIG. 1 as part of operating information that will incorporate with user cache. And the text in Column 6-10 includes with boot firmware, processor, memory elements (Column 7, started at line 50, for operating system information), preload parameters, request data obtained from mass storage, system directory, cache directory (started at line 55 in Column 9 to line 27 in Column 10), are associated with FIG. 6. All of these disclose **broad means**, "operating system **information**", and "installation **information**" (emphasis added).
- Regarding Claims 4, 9, and 14, Teoman shows FIG. 6, with blocks 108, 110, 112 used to access derivers directly to FILE 15 in user cache/mass storage. FILE 15 or data in mass storage is shown as AutoCAD, WINwork (shown in FIG. 11), or a particular application provided by a commercial sector connected by Networks server 29A, 29B (shown in FIG 1). Therefore, this data or FILE 15 by itself discloses broad means "a part number".

All Applicants arguments to Claims 3, 8, and 13 and 4, 9, and 14 in the Remarks have been considered. Applicants once address "ex parte Levy" and "In re Robertson (Remarks, page 7, lines 9-18). However, in this case, the Applicants' request for an extrinsic evidence would be improper since the limitations recited in these Claims simply are non-functional descriptive materials. The Examiner's illustration above could teach such broad and non-functional limitations.

Regarding Applicants' arguments to Claims 5, 10, and 15 (Remarks, page 7, started at line 19 to page 8).

As pointed out by Applicant that Teoman does not disclose *retrieving software associated with said identified one or more software element objects based on a part number*" (Remarks, page 8, lines 10-13), the rejection of Claims 5, 10, and 15 based on the art of record, Teoman, is withdrawn. See further details in Allowable Subject Mater.

Claim Rejections - 35 USC § 102

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

4. Claims 1-4, 6-9, 11-14 are rejected under 35 U.S.C. 102(e) as being anticipated by Teoman et al. (US No. 6,370,614 B1).

Given the broadest reasonable interpretation of followed claims in light of the specification.

As per Claim 1: Teoman discloses, "*A method for creating a preload, wherein an object of said preload is an aggregation of one or more software element objects, comprising the steps of:*

defining a particular preload object with one or more attributes (See FIG. 6, and associated text, particularly in column 10, lines 63-66. See FIG. 10 and associated text, referring to the area in FIG. 10 shown with "Preloading");

comparing attributes of said one or more software element objects with said one or more attributes of said particular preload object (See column 14, lines 61-66, "periodically compared"; and particularly, in column 12, lines 18-21, "user cache drivers compares the blocks"), *wherein each of said* *one or more software element objects constitutes one or more of a device driver object, an operating system object and an application software object* (See FIG. 6 and associated text)*;*

identifying one or more of said one or more software element objects whose attributes comprise said one or more attributes of said particular preload object (See FIG., pointing to FILE 15, and see FIG. 10, provided with check boxes in Preloading area, and see FIG. 11, with 'path')*; and*

installing software associated with said identified one or more software elements objects onto a particular preload associated with said particular preload object" (See column 15, lines 34-61, "to begin commanded preloads of the directory contents"; see column 16, lines 1-37, "to be preloaded into the user cache"; and see column 8, lines 65-67, "Once data from remote database is cached in user cache").

As per Claim 2: Teoman discloses, "*The method as recited in claim 1 further comprising the step of: modifying an attribute of said identified one or more software element objects to match said one or more attributes of said particular preload object*" (See FIG. 6 has means for identifying FILE 10. See FIG. 10 shows extensions of matching files could be changed. See in column 14, started at lines 59 to column 15, line 7, user would be alerted for correcting duplicated data).

As per Claim 3: Teoman discloses, "*The method as recited in claim 1, wherein each of said one or more software element objects is associated with attribute data, wherein said attribute data comprises one or more of an operating system information and an installation information*" (See FIG. 6, particularly, see: started at line 56, column 6 to line 27, column 10).

As per Claim 4: Teoman discloses, "*The method as recited in claim 1, wherein each of said one or more software element objects is associated with attribute data, wherein said attribute data comprises a part number* (See FIG 6, for example: Attribute in 111 is pointed to Device Driver 47, where *part number* is in Device Driver 47).

As per Claims 6-9: Claims are claiming a computer product that has claimed functionality corresponding to the method recited in Claims 1-4, respectively. Claims 6-9 are rejected in the same reasons as set forth in connecting to the rejections of Claims 1-4.

As per Claims 11-14: Claims are claiming a system that has claimed functionality corresponding to the method recited in Claims 1-4, respectively. Claims 11-14 are rejected in the same reasons as set forth in connecting to the rejections of Claims 1-4.

Allowable Subject Matter

5. Claims 5, 10, and 15 are objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

As pointed out by Applicant in the Remarks (page 8, lines 10-13) and Teoman, prior art of record, does not explicitly address limitation comprising at least feature, "*retrieving software associated with said identified one or more software element objects based on said one or more part numbers*", as recited in dependent Claim 5 and in such manners in dependent Claims 10, and 15.

Conclusion

6. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Ted T. Vo whose telephone number is (571) 272-3706. The examiner can normally be reached on 8:00AM to 5:30PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3694. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Ted T. Vo
Primary Examiner
Art Unit 2122
February 28, 2005